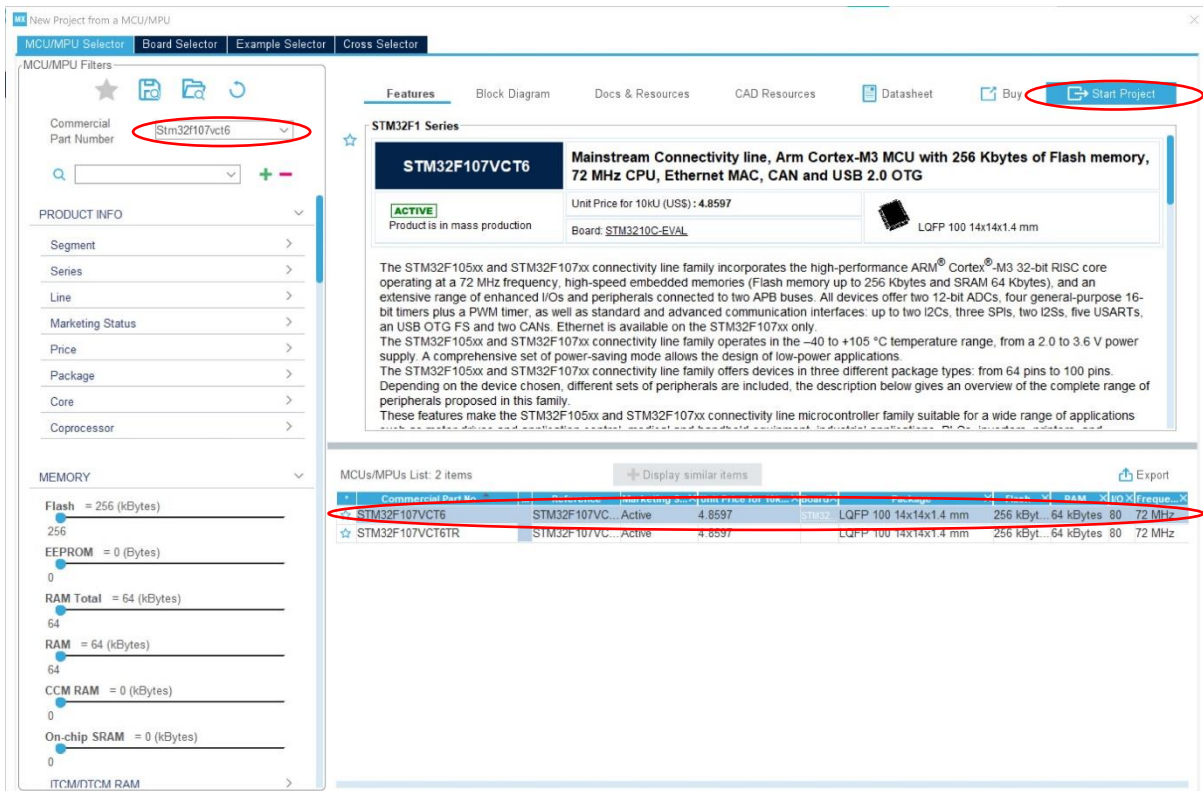
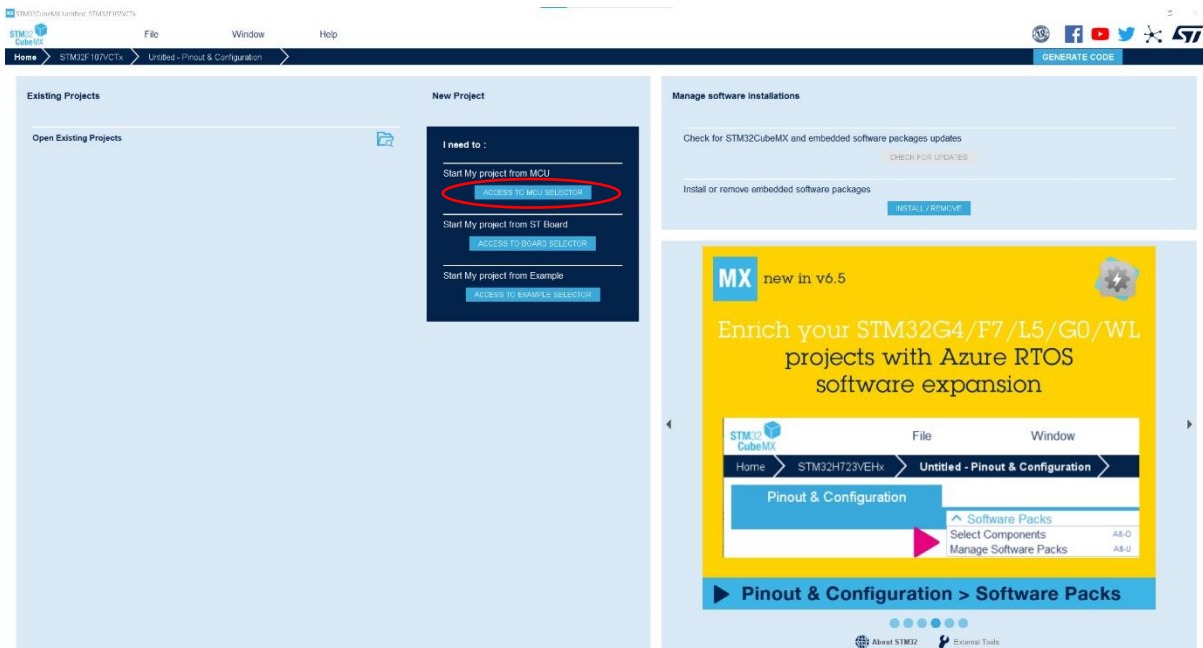


STM32F107VCT6 ICMP-Verbindung Grundkonfiguration mit CubeMX.

1. MCU wählen

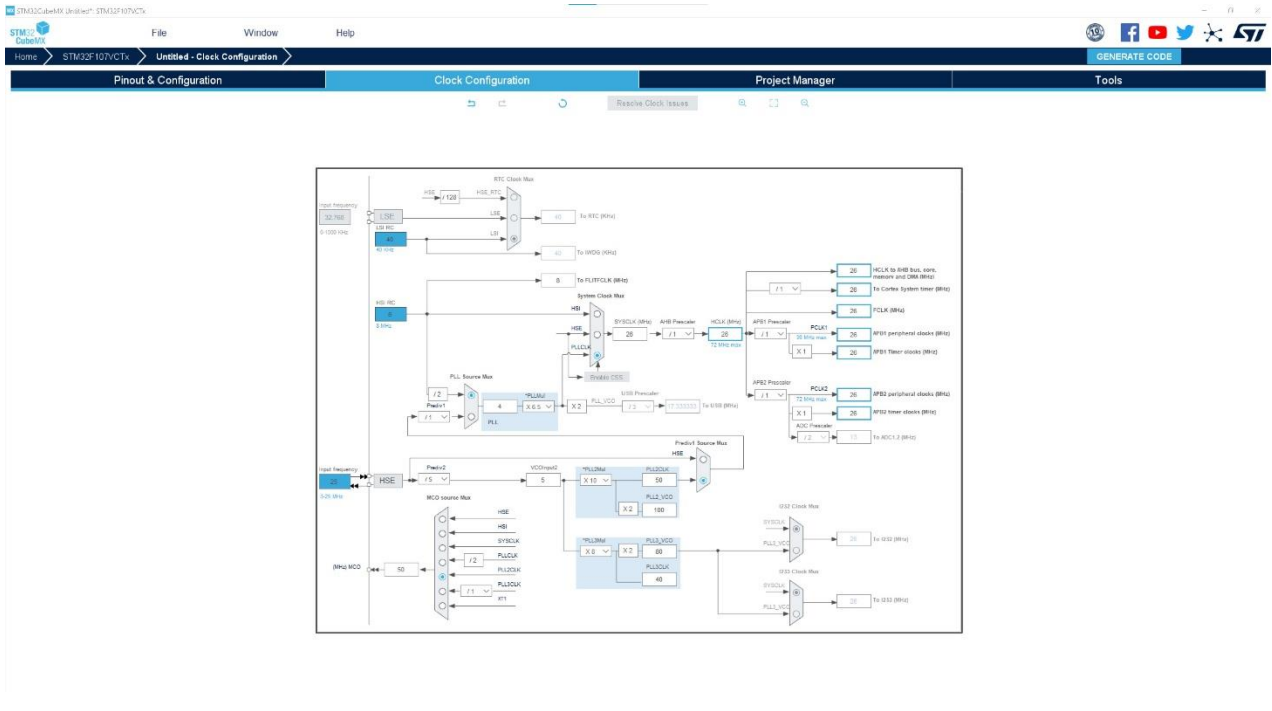


2. Master Clock Output und High Speed Clock aktivieren

The screenshot displays the STM32CubeMX software interface. The top navigation bar includes 'File', 'Window', and 'Help' menus, along with social media icons and the ST logo. The main workspace shows a pinout configuration for an STM32F107VCTx microcontroller. A sidebar on the left lists various system components, with 'RCC' highlighted in red. The central area features a 3D model of the microcontroller chip with its pinout labeled. Below the workspace, the 'RCC Mode and Configuration' window is open, showing the following settings:

RCC Mode and Configuration	
Mode	
High Speed Clock (HSE)	Crystal/Ceramic Resonator
Low Speed Clock (LSE)	Disable
<input checked="" type="checkbox"/> Master Clock Output	

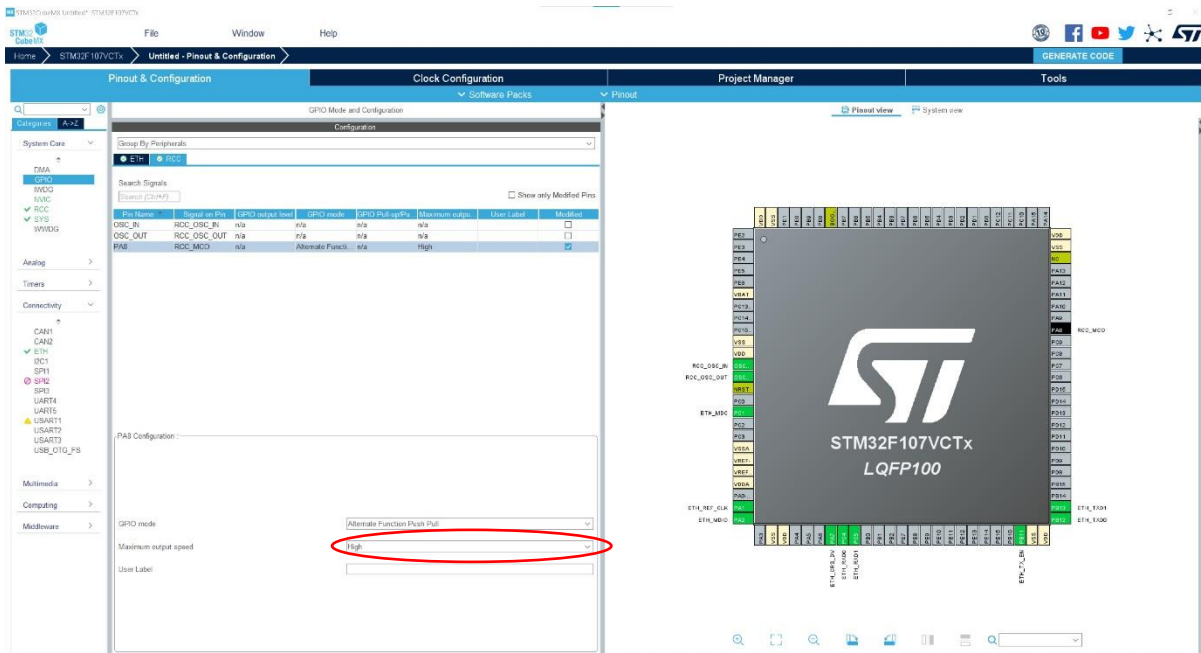
3. Clocks wie im Bild gezeigt konfigurieren



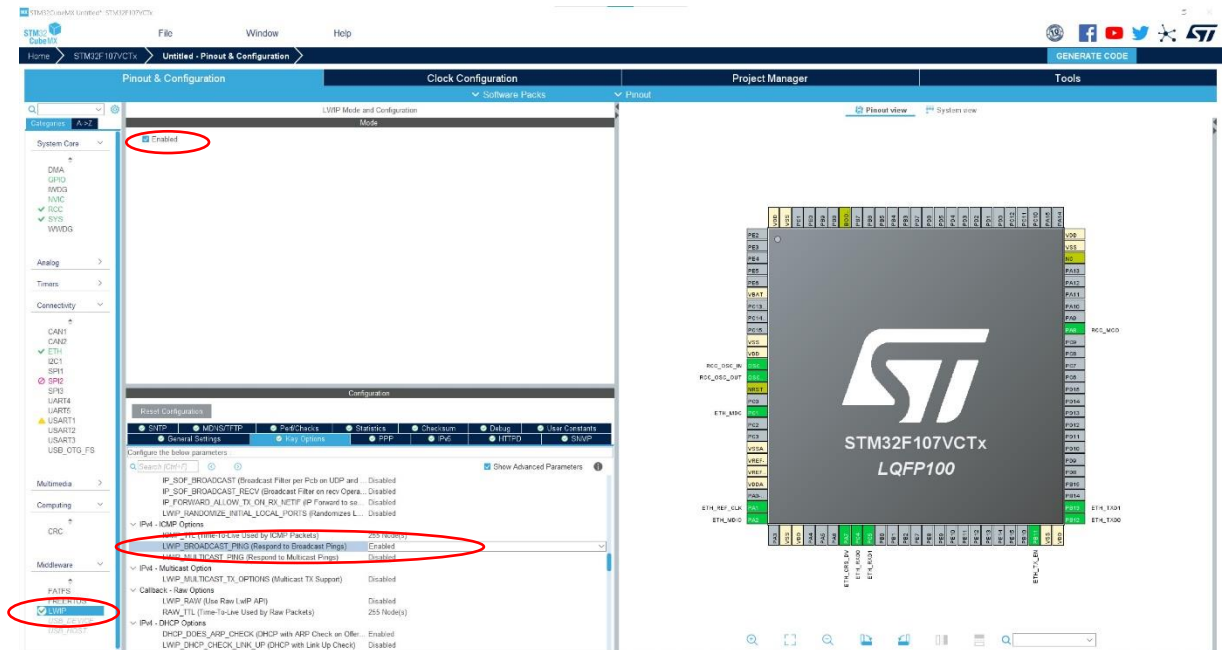
4. ETH aktivieren, PHY – Address -> 0

The screenshot shows the STM32CubeMX Pinout & Configuration interface. The 'Pinout' tab is selected, and the 'General' configuration page is displayed. The 'General' configuration page shows the 'PHY Address' set to 0. The 'Advanced' configuration page is also visible, showing the 'Ethernet Media Configuration' section. The 'Advanced' configuration page shows the 'Ethernet Media Configuration' section with the following settings: 'Advanced' checked, 'Ethernet Media Configuration' checked, 'Auto Negotiation' enabled, 'Speed' set to 100 Mbps, 'Duplex Mode' set to Full Duplex, 'General' checked, 'Ethernet Configuration' checked, 'PHY Address' set to 0, 'Ethernet Basic Configuration' checked, 'Rx Slope' set to Polling Mode, and 'Tx IP Header Checksum Computation' set to By hardware. The 'PHY Address' is circled in red. The 'Advanced' configuration page also shows the 'Ethernet Media Configuration' section with the following settings: 'Advanced' checked, 'Ethernet Media Configuration' checked, 'Auto Negotiation' enabled, 'Speed' set to 100 Mbps, 'Duplex Mode' set to Full Duplex, 'General' checked, 'Ethernet Configuration' checked, 'PHY Address' set to 0, 'Ethernet Basic Configuration' checked, 'Rx Slope' set to Polling Mode, and 'Tx IP Header Checksum Computation' set to By hardware.

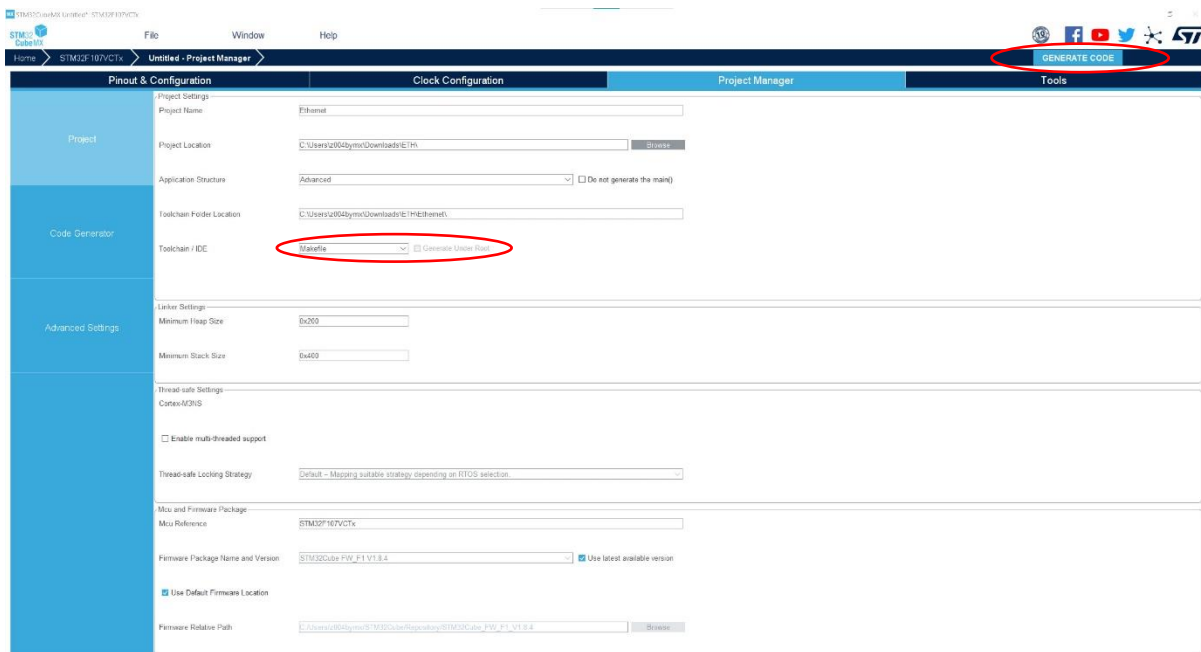
5. GPIO -> RCC -> PA8: Maximum output speed auf "High" stellen



6. LWIP (Leighweight IP) aktivieren, ICMP Broadcast und Multicast Ping unter „Key Options“ aktivieren



7. Unter „Project Manager“ Toolchain / IDE -> Makefile wählen, und Code generieren



8. Mainfile anpassen

CubeMX generiert ein Projektverzeichnis, welches nun von der gewünschten IDE (wir verwenden VS Code) genutzt werden kann. Der generierte Code genügt, um einen Link herzustellen (Windows zeigt in den Adapteroptionen „undefined Network“). Damit das Board eine IP-Adresse erhält, muss main.c wie folgt angepasst werden:

```
56 /* USER CODE BEGIN 0 */
57 extern struct netif gnetif;
58 /* USER CODE END 0 */

94     while (1) {
95         /* USER CODE END WHILE */
96         ethernetif_input(&gnetif);
97         sys_check_timeouts();
98         /* USER CODE BEGIN 3 */
99     }
100 /* USER CODE END 3 */
```

Das Board erhält nun eine IP-Adresse vom DHCP-Server. Eine manuelle IP-Konfiguration lässt sich bei CubeMX unter LWIP -> General Settings einstellen. Hierzu muss zuerst DHCP deaktiviert werden.

